

哈希表与跳表性能比较

学生姓名： 杨景凯

学 号： 520021910550

2022 年 3 月 1 日

目录

| | | |
|----------|----------------|----------|
| 1 | 理论部分 | 3 |
| 1.1 | 选择 | 3 |
| 1.2 | 分析 | 3 |
| 2 | 实验部分 | 3 |
| 2.1 | 实验说明 | 3 |
| 2.2 | 实验结果 | 3 |
| 2.3 | 实验结论 | 4 |

1 理论部分

1.1 选择

我选择跳表。

1.2 分析

跳表是在线性结构基础上进行发展的数据结构，它一方面依据随机层数的特性，能够快速查找到数据，另一方面维持了底层线性的特点，更适合于连续数据的遍历。

2 实验部分

2.1 实验说明

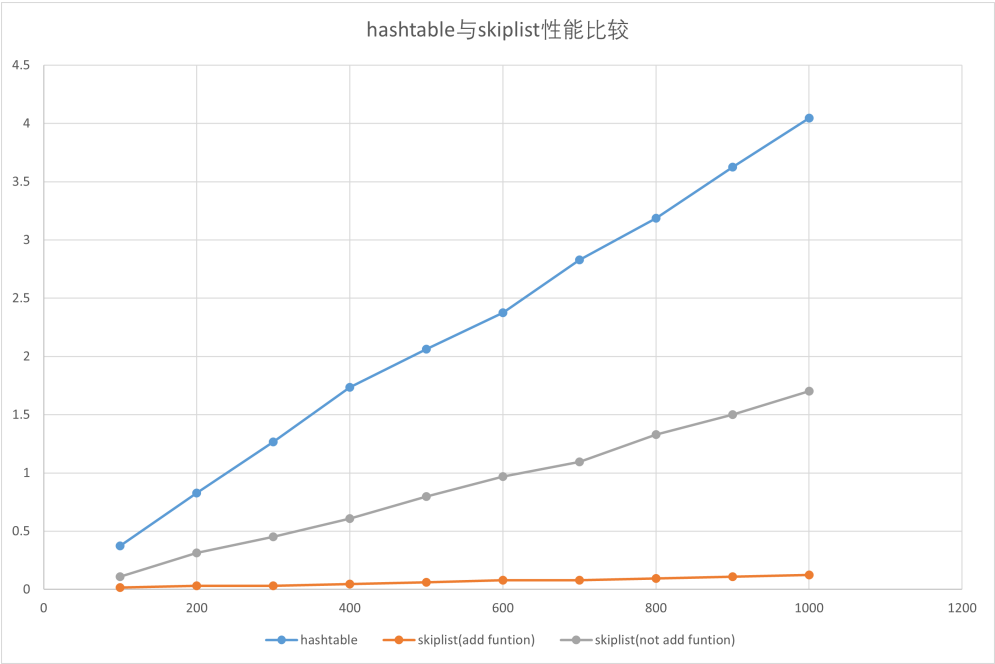
为了证明我的结论，我设计了如下实验。我比较了哈希表、未添加连续搜索函数（只使用简单搜索函数）的跳表和添加了连续搜索函数（连续线性遍历区间元素）的跳表。实验分为 10 组，每组区间搜索元素分别为 100 至 1000 个。为了更明显地体现差距，将搜索反复进行 10000 次。输出三种数据结构处理的时间（以秒为单位）。

2.2 实验结果

实验结果如下表所示：

| Table of result | | | |
|-----------------|--------------|--------------------------|------------------------------|
| searchtimes | hashtable(s) | skiplist(add funtion)(s) | skiplist(not add funtion)(s) |
| 100 | 0.375 | 0.015625 | 0.109375 |
| 200 | 0.828125 | 0.03125 | 0.3125 |
| 300 | 1.26562 | 0.03125 | 0.453125 |
| 400 | 1.73438 | 0.046875 | 0.609375 |
| 500 | 2.0625 | 0.0625 | 0.796875 |
| 600 | 2.375 | 0.078125 | 0.96875 |
| 700 | 2.82812 | 0.078125 | 1.09375 |
| 800 | 3.1875 | 0.09375 | 1.32812 |
| 900 | 3.625 | 0.109375 | 1.5 |
| 1000 | 4.04688 | 0.125 | 1.70312 |

根据实验结果做出的折线图如下：



2.3 实验结论

根据上述实验，无论是未添加连续搜索函数（只使用简单搜索函数）的跳表还是添加了连续搜索函数（连续线性遍历区间元素）的跳表，在区间搜索性能上都比哈希表要好。因此之前的选择和理论分析正确。